Pertemuan 3

Shell dan Pemrograman Shell

Objektif:

1. Mahasiswa mengetahui apa yang dimaksud dengan “Shell”.
2. Mahasiswa mengenal jenis-jenis Shell.
3. Mahasiswa mengenal kegunaan dari Shell.
4. Mahasiswa menguasai pemograman di dalam Shell (Bourne dan Korn Shell).

P3.1 Teori

Definisi dan Manfaat Shell

Shell merupakan perangkat lunak yang berfungsi sebagai penerjemah perintah, berisikan sejumlah perintah (yang sering kita gunakan pada prompt shell) dimana dapat kita jadikan sebagai satu prosedur dan dapat disimpan pada satu file.

Manfaat yang dapat diperoleh dengan adanya shell adalah:

1. Menangani sebuah prosedur yang terdiri dari sejumlah perintah.
2. Membangun utilitas yang sangat bermanfaat bagi banyak pemakai.
3. Menyederhanakan sebuah perintah yang sering dipergunakan.

Ada beberapa jenis shell yang sering dipergunakan, di antaranya:

1. Bourne Shell, diciptakan Stephen R. Bourne.
2. C Shell, diciptakan Bill Joy.
3. Korn Shell, diciptakan David G. Korn.

Komentar

Komentar diawali dengan tanda # . Semua karakter sesudah tanda itu hingga akhir baris diperlakukan sebagai komentar.

Menuliskan Script Shell

Langkah – langkah untuk menuliskan script shell :

1. Gunakan editor seperti editor Vi
2. Setelah menuliskan script shell, set perintah untuk menjalankan script tersebut. Contoh : $chmod +x nama file script
3. Jalankan Script sesuai dengan shell yang digunakan.
   Contoh sintaks : **bash** nama script atau **sh** nama script

Variabel Shell

Pada linux terdapat dua variabel, yaitu :

1. ***System Variables***, diciptakan dan di-*maintained* oleh sistem linux. Jenis variabel ini didefinisikan berupa huruf capital

Tabel 3.1 System variables

System Variables	Arti
BASH=/bin/bash	nama shell yang dibuat
BASH_VERSION=1.14.7.(1)	versi shell
COLUMNS=80	no kolom pada layar
HOME=/home/vivek	home direktori
LINES=25	no baris pada layar
LOGNAME= mahasiswa	login dengan nama mahasiswa
OSTYPE=linux	tipe sistem operasi yang digunakan
PATH=/usr/bin:/sbin:/bin:/usr/sbin	path settings
PS1=[\u@\h\W]\$	prompt settings
PWD =/home/students/Common	direktori lembar kerja
SHELL=bin/bash	nama shell
USERNAME=vivek	username adalah orang yang sedang menggunakan PC tersebut

2. ***User Defined Variables (UDV)***, variabel yang di buat dan di-*maintained* oleh pemakai. Variabel ditulis dengan huruf kecil.

Fungsi variabel

- Digunakan untuk menyimpan suatu nilai (bilangan atau string).
- Nilai dalam suatu variabel dapat juga diatur agar isinya tidak dapat berubah.
- Setiap pemakai mempunyai variabel tersendiri. Perubahan variabel atau penambahan variabel pada seorang pemakai tidak mempengaruhi pemakai yang lain.
- Variabel yang diciptakan pada prompt shell akan hilang dengan sendirinya pada saat proses logout. Variabel yang dibuat pada shell script akan hilang pada saat eksekusi shell script berakhir.
- Dalam sistem operasi Linux, ada juga beberapa variabel bawaan yang telah disediakan. Variabel ini disebut juga variabel lingkungan, karena dipakai untuk mengontrol lingkungan dari pemakai. Di antaranya: PATH, HOME, CDPATH, MAIL, dan lain-lain.
- Selain variabel bawaan, seorang pemakai dapat juga mendefinisikan sendiri variabel. Aturan mengenai pemberian nama variabel adalah:

1) Nama variabel diawali dengan sebuah abjad atau karakter garis bawah (_), kemudian diikuti oleh sejumlah karakter (a-z A-Z 0-9 atau _).
2) Panjang variabel bebas.
3) Huruf kapital dan huruf kecil mempunyai makna yang berbeda.
4) Pemberian nilai pada sebuah variabel diantaranya dilakukan dengan pernyataan *assigment*, tanda sama dengan (=).
5) Jika nilai yang diberikan pada suatu variabel mengandung spasi, nilai tersebut harus diawali dan diakhiri tanda " atau '.

- Pada Linux, dikenal adanya variabel lokal (variabel yang terdapat pada script anak, yaitu suatu shell script yang dipanggil oleh shell script yang lain) dan variabel global (variabel ekspor, pada script induk). Pemberian nilai pada variabel global di script anak tidak akan mempengaruhi nilai variabel tersebut di shell induk.
- Untuk melihat seluruh variabel yang telah didefinisikan pemakai, dapat digunakan perintah:

  set | pg atau **set | more**.
- Perintah **unset** digunakan untuk menghapus variabel.
- Pernyataan **readonly** sebelum nama variabel, mengakibatkan nilai pada variabel tersebut tidak dapat diubah.
- Membaca data dari papan ketik **(keyboard)** dan meletakkannya pada variabel menggunakan perintah **read**.

Nilai Exit

Semua perintah Linux menghasilkan nilai keluaran yang disebut sebagai nilai exit. Jika perintah berhasil dilaksanakan akan menghasilkan nilai ezit sama dengan nol, selain itu ada kesalahan dalam menjalankan perintah tersebut. Nilai exit tersebut dapat diketahui dengan menggunakan variabel **$?**.

Test

Perintah ini berfungsi untuk memeriksa kebenaran dari suatu ungkapan. Ungkapan itu dapat berupa bilangan, string, file dan juga operator. Pada Bab 4 disajikan macam-macam ungkapan yang mungkin dilakukan.

P3.2 Contoh Kasus

Langkah – langkah untuk menuliskan script shell dan menjalankan di editor vi adalah sebagai berikut :

1. Buka terminal di linux kalian
2. Buka editor vi di terminal dengan mengetikkan vi (spasi) (nama file). Misal disini saya buat $vi for
3. Lalu anda akan masuk ke dalam editor vi. Pertama masuk ke editor vi anda akan berada dalam mode perintah sedangkan untuk menuliskan script di editor vi, anda harus berada dalam mode penyisipan. Untuk itu ketikkan i untuk masuk ke mode penyisipan.
4. Setelah masuk mode penyisipan, anda ketikkan script shell yang anda ingin tulis.
5. Untuk save dan keluar dari editor vi dan kembal ke terminal, anda harus ke mode perintah. Untuk itu tekan tombol ESC lalu ketik :wq untuk menyimpan dan keluar dari editor vi seperti gambar di bawah ini. Disini kita akan membuat suatu perulangan pada pemrograman shell linux dengan menggunakan perintah for. Syntax for adalah berikut ini :

   for NAME [in WORDS ...]; do perintah; done

   Pada program ini kita menggunakan variabel angka yang di dalamnya terdapat angka 1 2 3 4 5 6 7 8 9 10. Lalu untuk menampilkan hasil perulangannya menggunakan echo “angka=$angka” dimana syntax ini berarti akan menampilkan angka=perulangan 1-10 yang berada di dalam variabel angka.

```
putu@putu-ananta: ~
File Edit View Terminal Help
for angka in 1 2 3 4 5 6 7 8 9 10;
do
        echo "angka=$angka";
done
~
~
~
:wq
```

6. Setelah ketik: wq, maka anda akan kembali ke terminal, lalu di terminal untuk menjalankan script yang telah anda buat pada editor vi anda, ketikkan $chmod <spasi> nama file untuk melakukan pengesetan execute permission untuk script yang sudah dibuat. Artinya, kita akan menngubah mode perizinan file sehingga file tersebut diizinkan untuk di eksekusi. Untuk mengubah mode perizinan, gunakan perintah chmod. Dilanjutkan dengan ketik ./ <spasi> nama file untuk menjalankan script program dan hasil outputnya seperti gambar di bawah ini. Misalnya untuk program ini kita ketikkan

   $chmod 755 for

   $./for

```
putu@putu-ananta: ~
File Edit View Terminal Help
putu@putu-ananta:~$ vi for
putu@putu-ananta:~$ chmod 755 for
putu@putu-ananta:~$ ./for
angka=1
angka=2
angka=3
angka=4
angka=5
angka=6
angka=7
angka=8
angka=9
angka=10
putu@putu-ananta:~$
```

P3.3 Latihan

Buatlah shell script untuk memperoleh huruf abjad dengan menggunakan for!

Jawab:

Output yang dihasilkan :

```
Huruf=a
Huruf=b
Huruf=c
Huruf=d
Huruf=e
Huruf=f
Huruf=g
Huruf=h
Huruf=i
```

P3.4 Daftar Pustaka

Kadir, Abdul.*Pengenalan Unix dan Linux.*Penerbit Andi Yogyakarta.2002.

Sidik, Betha. *Panduan Bekerja dalam Lingkungan Unix dan Linux*, Penerbit Informatika. 2004.